Ausbildung Fachinformatiker Anwendungsentwicklung

Programmierübung

C/Python–Programmierung / Simulation

1 Aufgabe

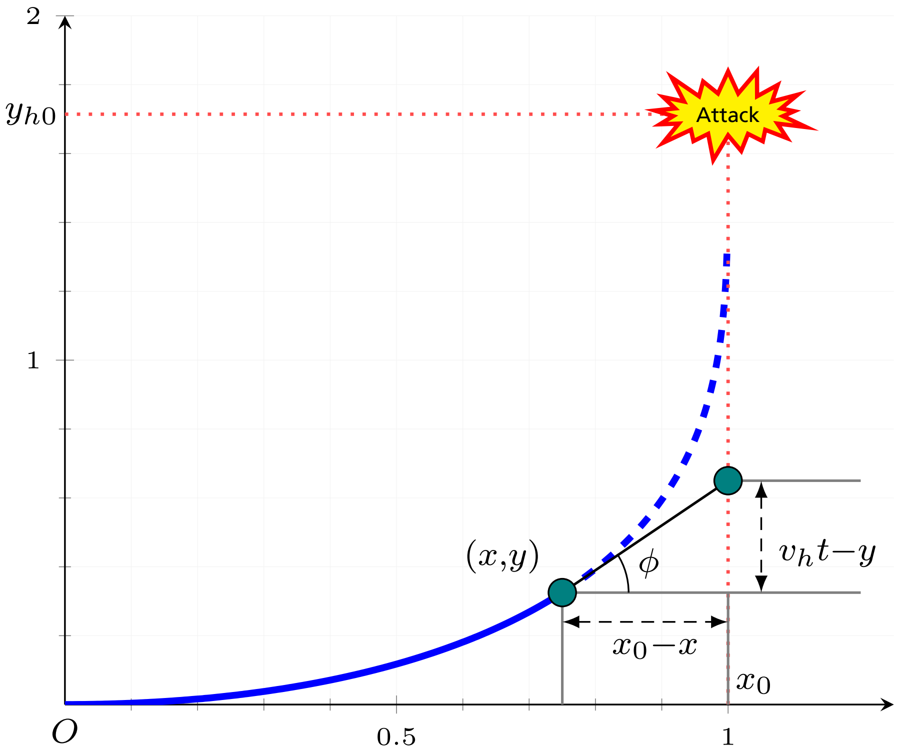
Es ist ein Programm (`pirate.exe` bzw. `pirate.py`) in der Programmiersprache C und der Programmiersprache `Python` zu schreiben, das folgendes Szenario lösen kann:

Ein Handelsschiff segelt mit konstanter Geschwindigkeit (v_h) in einem kartesischen Koordinatensystem vertikal in y–Richtung. Es befindet sich anfänglich an der Position $(x_0, 0)$ und sein Weg kann beschrieben werden durch die Kurve $x(t) = x_0$ und $y(t) = v_h t$.

Ein Piratenschiff an den Koordinaten $(0, 0)$ segelt zeitgleich mit dem Handelsschiff los und nimmt permanent Kurs auf die aktuelle Position des Handelsschiffs. Natürlich ist das Piratenschiff schneller, es hat die Geschwindigkeit $v > v_h$. D.h. $n = \frac{v_h}{v} < 1$. Wählt man also geeignete Einheiten, so kann man für die Berechnung setzen: $x_0 = x_h = 1$,$v = 1$ und $v_h = n < 1$ (Konkret zu Berechnung: $n = 3/4$).

Irgendwann wird das Piratenschiff das Handelsschiff eingeholt haben. Die Frage ist:

An welcher y–Position y_{h0} hat das Piratenschiff das Handelsschiff eingeholt. Da das Handelsschiff mit konstanter Geschwindigkeit segelt, ist die Frage natürlich äquivalent zu der Frage, zu welcher Zeit $t_{h0} = \frac{y_{h0}}{v_h}$.



2 Vorgehensweise

Simulieren Sie in kleinen Zeitschritten Δt die Bewegung beider Schiffe. Sie bekommen durch Wahl des geeigneten Zeitschritts ein hinreichend genaues Ergebnis.

Die beiden Schiffe treffen aufeinander, wenn der Abstand beider Schiffe sowohl in x–Richtung als auch in y–Richtung kleiner als ϵ ist. Dabei ist ϵ geeignet zu wählen. Die Wahl von ϵ und Δt sind voneinander abhängig, denken Sie darüber nach.

Zur Erleichterung wurde schon die notwendige Tabelle erstellt. Diese ist folgendermaßen von links nach rechts und oben nach unten zu lesen.

1. Für ein bestimmtes t (also auch k) werden die Position für das Handelsschiff $P_h = (x_h, y_h)$ und die Positionen des Piratenschiffes $P = (x, y)$ berechnet.
2. Es wird der (Kurs–)Winkel zwischen dem Handelsschiff und dem Piratenschiff aus den Abständen mit Hilfe des $\arctan()$ berechnet.
3. Mit dem Kurswinkel kann nun das jeweilige Δx und Δy berechnet werden, die das Piratenschiff im nächsten Zeitintervall vorwärts segelt.

k	t	x_h	y_h	x	y	$\tan\phi$	Δx	Δy
0	0	1	0	0	0	0	Δt	0
1	$1\Delta t$	1	$1n\Delta t$	$x^{(0)} + \Delta x^{(0)}$	$y^{(0)} + \Delta y^{(0)}$	$\frac{y_h^{(1)} - y^{(1)}}{1 - x^{(1)}}$	$\Delta t \cos\phi^{(1)}$	$\Delta t \sin\phi^{(1)}$
2	$2\Delta t$	1	$2n\Delta t$	$x^{(1)} + \Delta x^{(1)}$	$y^{(1)} + \Delta y^{(1)}$	$\frac{y_h^{(2)} - y^{(2)}}{1 - x^{(2)}}$	$\Delta t \cos\phi^{(2)}$	$\Delta t \sin\phi^{(2)}$
3	$3\Delta t$	1	$3n\Delta t$	$x^{(2)} + \Delta x^{(2)}$	$y^{(2)} + \Delta y^{(2)}$	$\frac{y_h^{(3)} - y^{(3)}}{1 - x^{(3)}}$	$\Delta t \cos\phi^{(3)}$	$\Delta t \sin\phi^{(3)}$
				$\dots$				
k_0	$k_0\Delta t$	1	$k_0 n\Delta t$	$x^{(k_0-1)} + \Delta x^{(k_0-1)}$	$y^{(k_0-1)} + \Delta y^{(k_0-1)}$	$\frac{y_h^{(k_0)} - y^{(k_0)}}{1 - x^{(k_0)}}$	$\Delta t \cos\phi^{(k_0)}$	$\Delta t \sin\phi^{(k_0)}$

Die Iteration kann beendet werden wenn

$$|x_h - x| < \epsilon \qquad \text{und} \qquad |y_h - y| < \epsilon$$

Die Zeit t_{h0} und die Entfernung y_{h0} ergeben sich dann aus den Werten $k_0\Delta t$ und $k_0 n\Delta t$.

Simulieren Sie dieses Problem sowohl mit der Programmiersprache `Python` als auch in der Programmiersprache `C`. Versuchen Sie verschiedene Werte von Δt und ϵ. Welche Beobachtungen machen Sie beim Testen?

3 Bewertung

Ziel ist ein möglichst exakter Wert des Ergebnisses. Beobachtungen während der Entwicklung und während des Testens sollen in einer kurzen Dokumentation niedergeschrieben werden (1–2 Seiten).

4 Abgabe und Zeitrahmen

Der Zeitrahmen dieser Aufgabe ist mit etwa 24 Stunden angesetzt.